

# Relative performance of ancilla verification and decoding in the $[[7,1,3]]$ Steane code

Ali Abu-Nada,<sup>1</sup> Ben Fortescue,<sup>1</sup> and Mark Byrd<sup>1,2</sup>

<sup>1</sup>*Department of Physics, Southern Illinois University, Carbondale, IL 62901, USA*

<sup>2</sup>*Department of Computer Science, Southern Illinois University, Carbondale, IL 62901, USA*

(Dated: May 22, 2013)

While ancilla post-selection is a common means of achieving fault-tolerance in quantum error-correction, it can lead to additional data errors due to movement or hold operations, and alternatives to post-selection may achieve lower overall failure rates due to avoiding such errors. We present numerical simulation results comparing the logical error rates for the fault-tolerant  $[[7,1,3]]$  Steane code using the technique of ancilla verification vs. the newer method of ancilla decoding, as described in [1]. We simulate a realistic QEC procedure in which failed ancilla creation requires storing the data until a new ancilla can be created; we find that the decoding method, which avoids the need for such storage, has an advantage when the failure probability is sufficiently high. We additionally analyze the effect of different classes of physical error (initialization, measurement, hold etc.) on the relative performance of these two methods.

## I. INTRODUCTION

### A. Ancilla postselection in quantum error correction

Quantum error-correcting codes (QECCs) provide a means to protect quantum data against noise, by encoding quantum states into larger Hilbert spaces such that some class of error operations are correctable [2, 3]. In a physically-realistic system, however, one must take into account that the correction operations themselves must be implemented using imperfect quantum operations. This leads to the common requirement that for quantum error correction (QEC) operations to be effective in protecting the data, they (and other operations on the data) must be implemented in a fault-tolerant way [4] i.e. in a way such that a single faulty quantum gate cannot lead to multiple errors on the data.

Some logical operations are naturally fault-tolerant, such as transversal gates in which every physical qubit is acted on by a separate gate, and the absence of any "cross-talk" between different physical qubits in an encoded block means there is no opportunity for errors to spread. In other cases, however, ancillary states which interact with the data as part of the QEC process need to be created using non-fault-tolerant circuits, and fault-tolerance is instead enforced by post-selection (ancilla verification), in which only those ancillas which satisfy some measurement outcome after being created are subsequently used to interact with the data [5, 6]. Thus it is not known beforehand whether a given ancilla will be used and one may need to perform multiple ancilla creations before obtaining one which passes post-selection.

This raises the question of how to ensure that a post-selected ancilla is available for QEC with sufficiently high probability so as not to significantly increase the overall failure rate of the QEC. Two obvious approaches are to either create ancillas sequentially until post-selection is passed, or to create many ancillas in parallel so there is a high probability that at least one will pass [7]. Both

approaches have somewhat analogous disadvantages: if an ancilla fails, sequential creation requires the data to wait (and, in general, accumulate errors) while a new ancilla is created. Parallel creation avoids this, but has the problem that in a given physical architecture there may be a very limited number of ancillas that can be created close to the data. Thus, a passed ancilla may be created some distance away and need to be moved into contact with the data, both data and ancilla accumulating errors while this occurs.

In this paper we consider addressing this problem using *ancilla decoding* [1], a technique originally devised to address the different problem of slow qubit measurements. As we discuss below, ancilla decoding removes the need for post-selection and thus guarantees that any created ancilla can be used with the data. Hence this ancilla can be created next to the data and QEC performed without any additional movement or waiting. Depending on the circuit layout and gate errors and timings, this may result in lower overall logical failure rates and hence in the case of a computation using a concatenated QECC, fewer resources required to achieve a given logical error rate.

### B. The Steane code and ancilla creation

In this work we compare the logical error rate  $P_L$  obtained when performing fault-tolerant QEC operations for the well-known  $[[7,1,3]]$  Steane code [3], performing either ancilla verification or decoding. The Steane code encodes a single logical qubit into the state of seven physical qubits, and can correct any error on a single qubit. It is a CSS stabilizer code, whose list of stabilizer generators is given in Table II in the appendix. In our analysis we consider the Steane ancilla technique, in which the code stabilizers are measured by copying error information to ancillas in encoded logical states.

A (non-fault-tolerant) picture of part of the QEC process is shown in Figure 1, in which the data (qubits  $|qd\rangle$ ) interacts with a prepared logical  $|0_L\rangle$  (where subscript  $L$  denotes logical states) ancilla state (qubits  $|a\rangle$ ) to de-

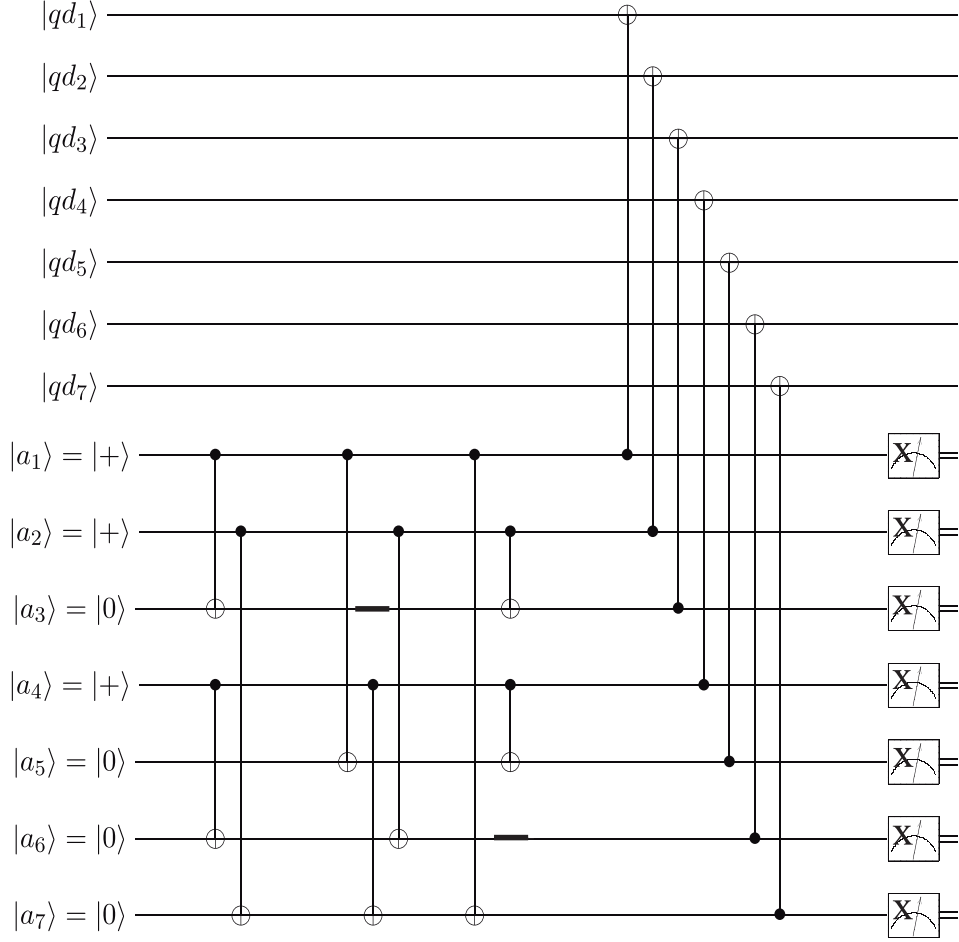


FIG. 1. A Steane code circuit with non-fault-tolerant Steane syndrome extraction, where  $|a_1\rangle, |a_2\rangle, \dots, |a_7\rangle$  are the ancilla qubits. These ancilla will interact with each other via CNOT gates to produce the logical ancilla state (*i.e.*  $|0_L\rangle$ ). Also,  $|qd_1\rangle, |qd_2\rangle, \dots, |qd_7\rangle$  are the data qubits. The ancilla-data interactions are implemented by a set of transversal CNOT gates, where any  $Z$  error in the data will propagate to the ancilla. Ultimately, the ancilla will be measured in the  $X$ -basis to determine the error syndrome.

termine the error syndrome for  $Z$  errors (an analogous interaction occurs with a logical  $|+_L\rangle$  to correct for  $X$  errors). We see, for example, that for fault-tolerance the  $|0_L\rangle$  ancilla, which interacts with the data as the source for a transversal CNOT gate, needs to be prepared so that a single gate failure will not cause the ancilla to have multiple Pauli  $X$  errors (likewise with the  $|+_L\rangle$  state and  $Z$  errors). Such errors would get transferred to the data, and can occur in the circuit shown in Figure 1, since the CNOT gates involved in preparing  $|0_L\rangle$  can propagate a single gate error to multiple qubits within the ancilla block.

The standard approach for ensuring this using ancilla verification (which occurs between ancilla preparation and data interaction) is shown in Figure 2. Passing the verification procedure is dependent on the outcome of the transversal  $Z$  measurement performed on the verifier. An analogous procedure occurs for  $Z$  errors and the  $|+_L\rangle$ .

The alternative method of ancilla decoding [1] is il-

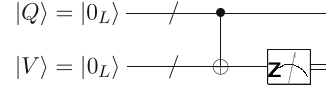


FIG. 2. An example of a single ancilla preparation and verification circuit taken from the Steane code with Steane syndrome extraction. Where  $|Q\rangle$  is the encoded ancilla state, and,  $|V\rangle$  is the encoded verifier state. Any  $X$  errors which have occurred in the creation of  $|Q\rangle$  will propagate to the verifier via the transversal CNOT.

lustrated in Figure 3. The basic principle is that after interacting the ancilla with the data, one applies a decoding operation to the ancilla to extract both the error syndrome (as usual) and to determine the presence of any errors on the ancilla which may have been transferred to the data ( $X$  or  $Z$ , depending on the ancilla in question). In addition, the ancilla interacts with a second ancilla block which is in a product of  $|0\rangle$  states (*i.e.*

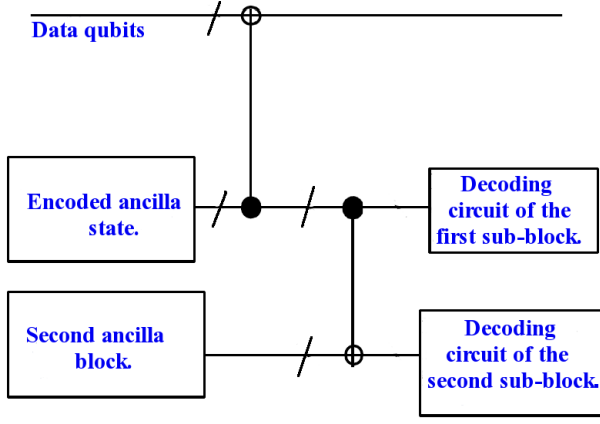


FIG. 3. The ancilla decoding procedure. After interacting the data with a non-postselected ancilla a decoding procedure is applied to determine the error syndrome and any multi-qubit errors which may have been transferred to the data. A second block allows one to distinguish between errors from encoding and decoding

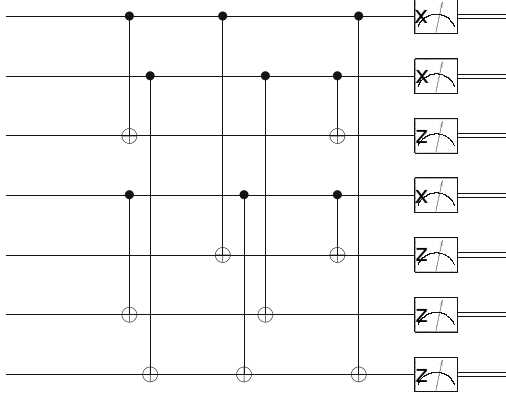


FIG. 4. A schematic for the decoding circuit of the non-postselected  $|0_L\rangle$  ancilla for the Steane code. We are able to detect single- or/and multi- $X$  qubit errors which propagate to the data during the ancilla-data interactions (there is an equivalent circuit for decoding the non-postselected  $|+_L\rangle$  ancilla to detect the single- or/and multi-  $Z$  qubit errors to the data). After the decoding, the ancilla will be measured in the  $X$ - and  $Z$ - basis as shown in the Figure. Measurements in the  $X$ - basis allow us to detect the  $Z$  errors which were initially in the data, while the  $Z$ - basis measurements will allow us to detect the  $X$  qubits errors due to the non-postselected ancilla.

$|0\rangle^{\otimes 7}$ ), which is also decoded after interacting with the data. This allows one to distinguish between errors of the first ancilla acquired during ancilla encoding (which will have been propagated to the data and need correcting) and those during decoding (which will not be transferred to the data or to the second block). The decoding circuit is illustrated in Figure 4.

Any propagated errors (single or multi-qubit) can then be corrected on the data. An important factor in the suc-

cess of this technique is that, for the purposes of fault-tolerance, only “first-order” patterns of errors, which can be caused by a single gate failure (though they may still affect multiple ancilla qubits through error propagation, or through errors on two-qubit gates) need to be corrected. Furthermore, many error patterns are equivalent up to stabilizer operations. This allows the limited information from the decoder to be sufficient to correct any such errors. Secondly, different classes of errors ( $X$  and  $Z$  or vice versa) are respectively detected by the standard syndrome measurement and by the additional decoder syndrome. For example, a  $|0_L\rangle$  ancilla is used to determine the syndrome for  $Z$  errors on the data, but is at risk of propagating  $X$  errors to the data, with the additional decoding giving information about the latter. The use of this method, therefore, means that any ancilla which is created (regardless of first-order errors) can be used in QEC, removing the need for verification. The encoding circuit is illustrated in Figure 9 in the appendix, each gate in that circuit being faulty and capable of producing an error which will propagate (either as single or multi-qubit error) to the data via the transversal CNOT gates (as shown in Figure 1). For example, if a single fault has been produced by the two output channels of CNOT-7, or -8, or -9 (in Figure 9), then this error will propagate as a two-qubit error to the data. This is discussed in more detail in the appendix.

As mentioned above, the original proposed advantage for ancilla decoding was to avoid long waits for ancilla verification in the case of slow measurements. Since, unless performing non-Clifford-group operations on our data, we can operate in the “Pauli frame” (merely classically recording the necessary corrections and updating the stabilizer accordingly rather than actually applying them), there is no corresponding need for the data to wait for the outcome of the ancilla decoding operations. One could also partially avoid the problem of waiting for verification by simply beginning the verification well in advance, but even for fast measurements, this does not avoid any delays due to having to restart preparation if verification fails. In our analysis we consider the case of measurement operations no slower than any other gate, and show that decoding still gives an advantage for this reason, even if verification failure is rare.

As discussed above, we are interested in the limitations imposed by practical QEC architecture, in which a verified ancilla may not always be available. We simulate a case where ancilla creation occurs serially and a failed ancilla verification requires the data to wait while a new ancilla is created and compare the performance (in terms of the overall  $P_L$ ) of a QEC procedure under these circumstances to QEC using the decoding method. While we consider the specific case of the Steane code using the Steane ancilla technique, it is conjectured [1] that a similar decoding procedure can be found for other CSS QECCs.

## II. SIMULATION PROCEDURE

To compare the two methods of ancilla interaction, we performed Monte Carlo simulations of the complete QEC procedure (so interaction of the data with two ancillas, one for the correction of each of  $X$  and  $Z$  errors) implemented using faulty gate operations. Our simulation software was QASM-P, software based on QASM by Cross [8]. All gates were simulated using the following common stochastic error model for depolarising noise, a function of a single error probability  $p$ :

1. Attempting to perform a data qubit identity  $I$  (hold operation), but instead performing a single qubit operation  $X$ ,  $Y$ , or  $Z$ , each occurring with probability  $p/3$ .
2. Attempting to initialize a qubit to  $|0\rangle$  or  $|+\rangle$ , but instead preparing  $|1\rangle$  and  $|-\rangle$ , respectively, with probability  $p$ .
3. Performing a  $Z$ -basis or  $X$ -basis qubit measurement, but reporting the wrong value with probability  $p$ .
4. Attempting to perform a CNOT gate, but instead performing a CNOT followed by one of the two-qubit operations  $I \otimes X$ ,  $I \otimes Y$ ,  $I \otimes Z$ ,  $X \otimes I$ ,  $X \otimes X$ ,  $X \otimes Y$ ,  $X \otimes Z$ ,  $Y \otimes I$ ,  $Y \otimes X$ ,  $Y \otimes Y$ ,  $Y \otimes Z$ ,  $Z \otimes I$ ,  $Z \otimes X$ ,  $Z \otimes Y$ , or  $Z \otimes Z$ , each with probability  $p/15$ .

We considered a range of values of  $p$  below the threshold for Steane code of roughly  $10^{-4}$  [9]. All gates are assumed to have the same duration, with hold operations implemented by applying the identity gate the appropriate number of times.

To determine  $P_L$  for the QEC procedure, the data was first prepared, without error, in a logical eigenstate  $|0_L\rangle$ . The QEC procedure was then performed using the error model above. Finally, a second QEC was performed with errors turned off, to project the output state into a logical eigenstate. This eigenstate was then compared against the input and a logical error reported if they did not match. The overall procedure is illustrated in Figure 5. From the Monte Carlo simulation we obtain  $P_L$  as a function of  $p$ . Note that due to the use of the  $|0_L\rangle$  state, only logical  $X$  errors are detected; this suffices to compare the techniques since by the symmetry of the underlying physical error model we expect the rate of logical  $Z$  errors to be approximately the same (up to some slight asymmetry due to performing one correction before the other).

When simulating ancilla verification, we assume a serial creation process for each of the two ancillas used in the QEC. We assume the ancilla is initially prepared an appropriate length of time prior to the QEC so that the data is not required to wait for ancilla preparation, so long as the first ancilla successfully passes verification. However, if ancilla creation fails, the data is held until a new ancilla can be created and verified. Holding the

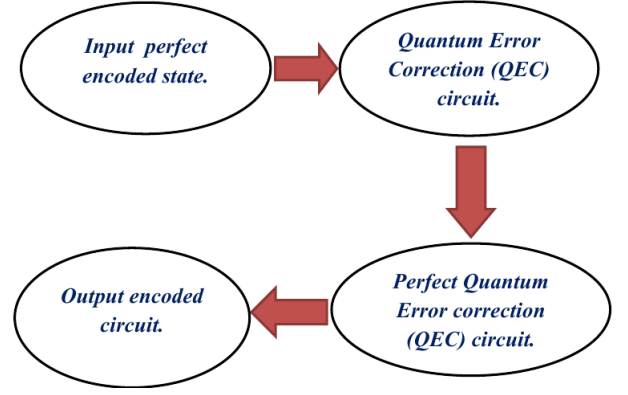


FIG. 5. Steps involved in the QEC simulation.

data continues until verification is passed, at which point the QEC continues. We hold the data for a total of 6 time steps (the number required to create and verify a new ancilla) for every failed verification, with no upper bound to the number of failures that can occur. Once an ancilla has successfully passed verification the QEC procedure continues. To determine the extent to which the verification process is affected by this data holding (as opposed to any other element), we additionally simulate an unrealistically optimistic case of verification where no additional holding is required if verification fails.

Additionally, we generated simulation failure rates in which particular classes of errors were considered in isolation i.e. with other errors turned off. As in e.g. [10], we define these classes of errors as follows:

- Class 0: errors from preparation and measurement.
- Class 1: errors from single-qubit gates (including the identity).
- Class 2: errors from two-qubit gates (i.e. the CNOT gate).

## III. RESULTS AND ANALYSIS

A direct comparison of the  $P_L$  for all three QEC techniques is shown in Figure 6. The results show an advantage for the decoding technique over verification over the whole range of errors considered, larger at larger error rates with a roughly 2-fold reduction as the maximum. Comparison with the  $P_L$  values for verification without additional holds, which are lower still, indicates as expected that the increased errors in the verification technique are due to ancilla failures, and that in the absence of such failures the additional operations required for ancilla decoding lead to a slightly higher error rate. The rate of ancilla failure as a function of  $p$  is plotted in Figure 7. Since ancilla verification detects single errors on the ancilla, this is roughly equal to (number of error locations in ancilla creation circuit)  $\times p$ , and the rate scales

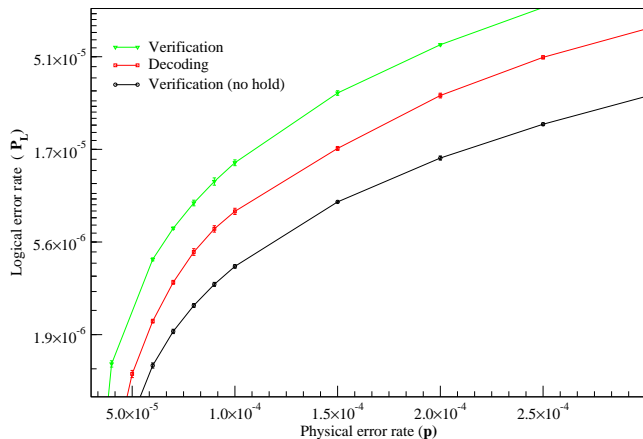


FIG. 6. Logarithmic error rate vs. physical error rate for the

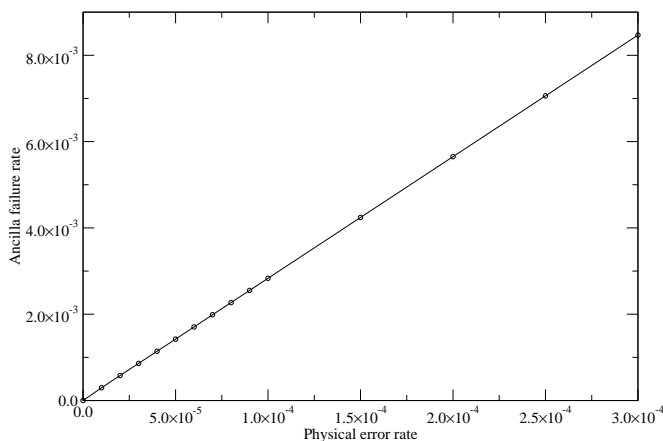


FIG. 7. The ancilla failure rate vs. physical error rate for the  $[[7,1,3]]$  Steane code with Steane ancilla

linearly with  $p$  as expected. Thus at low physical error rates ancilla failure occurs in a proportionally low fraction of QEC events, but still contributes significantly to the logical error rate.

Comparing the error rates by class in Figure 8 we see as expected that the largest source of error in verification is due to hold operations, while the other two techniques are dominated by CNOT errors. This emphasizes the need to take into account nondeterministic operations (such as verification) when assessing the impact of gate errors on QEC procedures.

#### IV. CONCLUSION

Using Monte Carlo simulations, we have compared the logical error rate,  $P_L$ , for implementations of the Steane code with Steane ancillas using two different QEC techniques: ancilla verification with serial recreation for failure events, and the ancilla decoding procedure. We find that, even when measurement times are no longer than

those for other operations and verification failures are rare, the decoding procedure, though otherwise more complex, is advantageous in avoiding data holds due to verification failures and leads to lower failure rates. Clearly these results are highly dependent on the error model chosen: many physical systems are dominated by errors in two-qubit operations, and a significantly lower error for holds might well result in verification being the preferred method. However, as well as showing that ancilla decoding can be of use even in the fast measurement regime, the results demonstrate that considering additional detail in practical implementations of QEC procedures (such as the need for ancilla recreation if verification fails) can make a significant difference to overall error rates and the preferred means of implementing a QEC. The particular case of verification is of interest given that many QECCs rely on post-selection to achieve fault-tolerance. We note that an analysis of the effect of decoding in trapped-ion-based quantum computing has recently been published[11].

One motivation for assuming a serial rather than parallel ancilla creation process in the verification model was that we were not required to make any particular assumptions about the layout of the underlying physical system. However, parallel creation, as a discussed in the introduction, is an obvious possible solution to avoid verification delays, and for a given layout it may well be that any additional errors due to qubit movement in the parallel case are less damaging than those incurred through serial creation, and possibly less than those from the additional operations in decoding. Comparing the three techniques over a range of likely layouts and error models would therefore be a promising avenue for future research.

#### ACKNOWLEDGMENTS

We thank Panos Aliferis and Russell Ceballos for helpful discussions. Supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D12PC00527. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

#### Appendix: Details of ancilla decoding

In this section we describe the possible ancilla errors due to a single fault in the encoding circuit, and the corresponding error syndromes when using the ancilla decoding technique. This is an expansion of the summary given in [1]. We will consider the case of  $X$  errors due



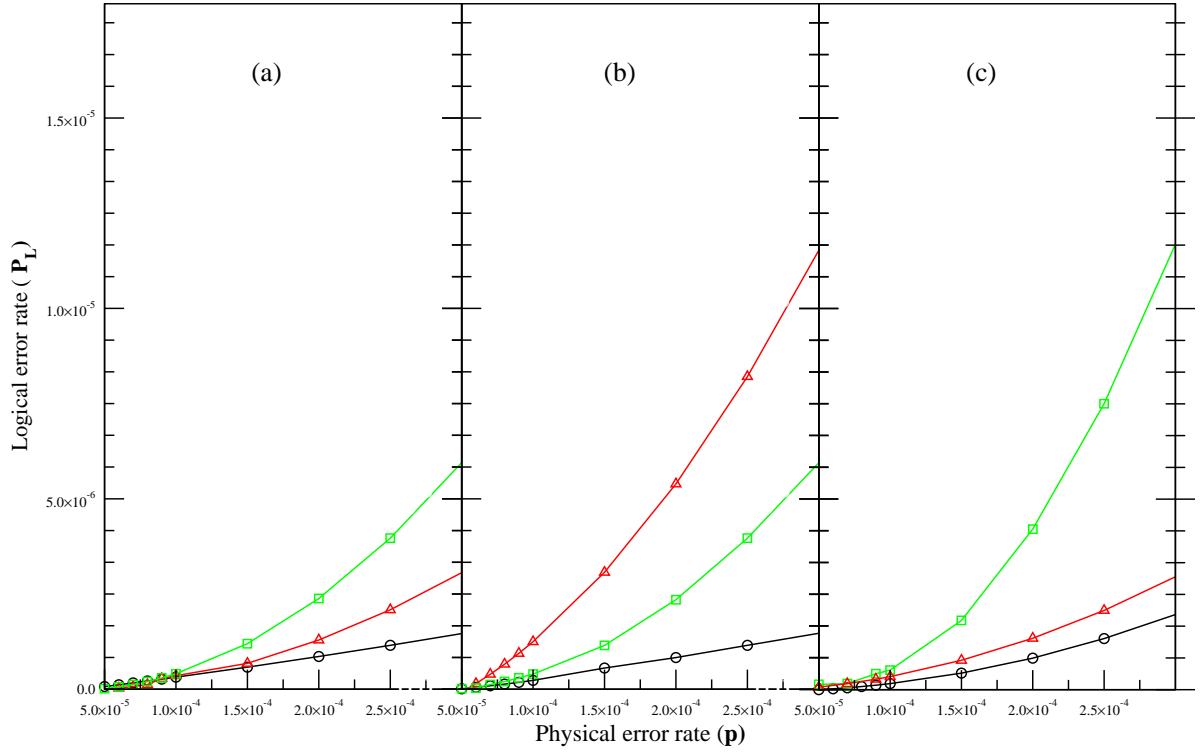


FIG. 8. Logical error rate  $P_L$  vs. physical error rate  $p$  for *class* – 0 errors (black circle points), *class* – 1 errors (red triangle points), and *class* – 2 (green square points) for (a) Verification (without additional holds), (b) Verification and (c) Decoding.

to faults in the encoding circuit for  $|0_L\rangle$ ; the case of  $Z$  errors when encoding  $|+\rangle$  is directly analogous.

Figure 9 illustrates the circuit for encoding  $|0_L\rangle$ , which possesses two types of gates: (a) single qubit gates defined as preparation of  $|0\rangle$  and  $|+\rangle$  with waiting gates represented by bold line segments in Figure 9, as well as (b) two qubit gates defined as CNOT gates. The decoding circuit can uniquely identify any single-qubit error. Two-qubit-errors caused by a single fault will either be errors on both outputs of a CNOT gate, or a single error which at some point propagates to two errors via CNOT, thus it suffices to further consider only the case of a CNOT gate on which both outputs have  $X$  errors. These are listed in Table I according to the encoding circuit labelling given in Figure 9.

As shown, dual output  $X$  errors propagate to single  $X$  errors to the data in the cases of CNOT gates 4, or 5, or 6; or as two  $X$  errors to the data in the cases of CNOT gates 7, or 8, or 9. See Table I. However, for CNOT gates 1, or 2, or 3, dual  $X$  errors in the two output channels will propagate as four  $X$  errors on the data. These are equivalent to the  $X$ -stabilizer generators of the Steane Code, which accordingly do not affect the data. The stabilizer generators of the Steane Code are listed in II. Finally, for CNOT gates 4, or 5, or 6; the single faults which are produced in the two output channels will propagate as a dual  $X$  errors on the data.

These  $X$  errors (single and double  $X$  errors) on the data can be identified by the decoding circuit (Figure 4)

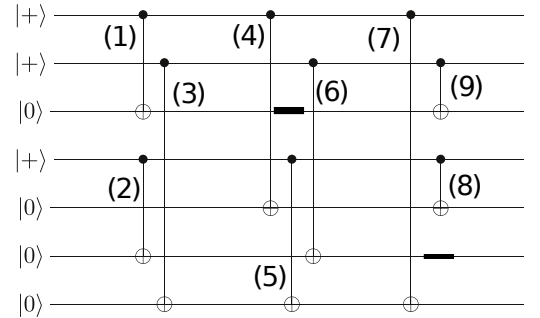


FIG. 9. Encoding circuit for  $|0_L\rangle$

TABLE I. Examples of output errors produced by  $X$  errors on both outputs of CNOT gates in the  $|0_L\rangle$  encoding circuit (Figure 9).

CNOT	Multi-qubit data errors
1	$X_1 X_3 X_5 X_7 \equiv G_1$
2	$X_4 X_5 X_6 X_7 \equiv G_2$
3	$X_2 X_3 X_6 X_7 \equiv G_3$
4	$X_1 X_5 X_7 \equiv G_1 X_3$
5	$X_4 X_5 X_7 \equiv G_2 X_6$
6	$X_2 X_3 X_6 \equiv G_3 X_7$
7	$X_2 X_3 \equiv G_3 X_6 X_7$
8	$X_4 X_5 \equiv G_2 X_6 X_7$
9	$X_1 X_7 \equiv G_1 X_3 X_5$

TABLE II. The stabilizer generators of the Steane code.

X-stabilizer	Z-stabilizer
$G_1 = X_1 X_3 X_5 X_7$	$G_1 = Z_1 Z_3 Z_5 Z_7$
$G_2 = X_4 X_5 X_6 X_7$	$G_2 = Z_4 Z_5 Z_6 Z_7$
$G_3 = X_2 X_3 X_6 X_7$	$G_3 = Z_2 Z_3 Z_6 Z_7$

and so corrected. As seen, every multi-qubit error, is, up to stabilizers, equivalent either to a single-qubit error or to one of three two-qubit errors,

- 
- [1] D. P. DiVincenzo and P. Aliferis, Phys. Rev. Lett. **98**, 020501 (2007).  
[2] P. W. Shor, Phys. Rev. A **52**, R2493 (1995).  
[3] A. M. Steane, Phys. Rev. Lett. **77**, 793 (1996).  
[4] J. Preskill, in *Introduction to Quantum Computation and Information*, edited by H.-K. Lo, S. Popescu, and T. P. Spiller (World Scientific, Singapore, 1998).  
[5] P. W. Shor, in *Proceedings of the Symposium on the Foundations of Computer Science* (IEEE Press, Los Alamitos, CA, USA, 1996) pp. 56–65.  
[6] A. M. Steane, Phys. Rev. Lett. **78**, 2252 (1997).  
[7] N. Isailovic, M. Whitney, Y. Patel,

- tel, and J. Kubiawicz, in *Proceedings of the 35th Annual International Symposium on Computer Architecture* (IEEE Computer Society, Washington, DC, USA, 2008) pp. 177–188.  
[8] A. Cross, “QASM,” <http://www.media.mit.edu/quanta/quanta-web/projects/qasm-tools> (2006).  
[9] A. M. Steane, Phys. Rev. A **68**, 042322 (2003).  
[10] A. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Phys. Rev. A **86**, 032324 (2012).  
[11] Y. Tomita, M. Gutiérrez, C. Kabytayev, K. R. Brown, M. R. Hutsel, A. P. Morris, K. E. Stevens, and G. Mohler, ArXiv e-prints (2013), arXiv:1305.0349 [quant-ph].